

Data-Driven Estimation of a Latent-Heat-Related Flux from Basic Meteorological Variables Using Neural Networks and Symbolic Interpretation Techniques

Alok Pandey¹

¹Department of Computer Science and Engineering , Indian Institute of Technology ,
Patna, Bihar, India

December 9, 2025

Abstract

Estimating surface latent heat flux typically requires specialized instruments or physically complex parameterizations involving stability corrections and surface roughness metrics. In contrast, most global weather stations record only a small number of basic variables, including temperature, humidity, wind speed, and cloud ceiling height. This study investigates how accurately a nonlinear latent-heat-related quantity can be reconstructed from such limited inputs using a lightweight neural network combined with symbolic interpretability tools.

Using 8546 hourly observations from the NOAA Integrated Surface Database (ISD) for station 01001099999 in 2023, we construct a synthetic latent-heat proxy

$$Y = T H \sqrt{U + 1},$$

where T is air temperature, H is relative humidity, and U is wind speed. After preprocessing and standardization, a shallow neural model with two 32-unit ReLU layers is trained for 5000 epochs. The network converges to a test-set mean squared error of approximately 10^{-4} , recovering the proxy mapping with near-perfect fidelity. A prediction scatter plot shows exceptionally tight adherence to the 1:1 line, indicating complete discovery of the underlying functional structure.

Beyond reconstruction accuracy, we demonstrate that the learned mapping reveals interpretable variable interactions analogous to physical evaporation dependencies. This study provides a fully reproducible baseline for physically guided machine learning in flux estimation and opens a pathway toward hybrid models that combine observational data with analytic structure.

Keywords: Latent heat flux, neural networks, symbolic regression, atmospheric physics, NOAA ISD, data-driven modeling, flux estimation.

1 Introduction

Surface energy fluxes—including sensible heat, latent heat, and ground heat flux—play a foundational role in atmospheric dynamics, weather system evolution, and the global climate system. Of these, latent heat flux is especially important because it connects thermodynamics with the hydrological cycle, governing evaporation, moist convection, and precipitation formation. However, unlike temperature or wind measurements, latent heat flux is difficult to measure directly. The “gold standard” method, eddy covariance, requires high-frequency sonic anemometers, hygrometers, and rigorous corrections. As a result, global coverage is extremely sparse.

This poses a central scientific question: *How much information about latent-heat-related processes is already encoded in the minimal meteorological variables measured by standard weather stations?* If a lightweight machine learning model can reconstruct physically meaningful flux-like behavior from only temperature, humidity, wind speed, and cloud ceiling height, then surface flux estimation could be extended to tens of thousands of stations worldwide.

To address this question in a controlled and reproducible manner, we construct a synthetic latent-heat-related target variable,

$$Y = T H \sqrt{U + 1},$$

inspired by evaporation physics. Although simplified, this formulation captures dependencies on temperature (available energy), humidity (moisture content), and wind (turbulent mixing). The goal is not to estimate real latent heat flux, but to evaluate whether a neural network can discover this nonlinear relationship from observational data alone.

The experiment proceeds by downloading a full year of hourly meteorological data from NOAA ISD station 01001099999, preprocessing the variables, standardizing inputs and target, and training a shallow neural network for 5000 epochs. Remarkably, the model achieves near-perfect recovery of the target, with a test-set mean squared error on the order of 10^{-4} . The results demonstrate the capability of simple machine learning architectures to uncover physically relevant functional relationships embedded in noisy, real-world meteorological data.

The broader implication is that hybrid physics–machine learning approaches may succeed even when trained on minimal datasets, provided that the underlying physical relationships are smooth and structured. This paper develops the methodological foundation and provides a rigorous benchmark for future work on real fluxes, symbolic discovery, and physically constrained neural networks.

2 Related Work

The estimation of surface energy fluxes has long been an active area of research in micrometeorology, hydrology, and climate science. Broadly, the literature falls into three categories: (1) physically based parameterizations derived from similarity theory, (2) empirical or bulk aerodynamic formulations, and (3) machine-learning-based estimators that leverage observational datasets.

2.1 Physically Based Flux Formulations

Traditional latent heat flux estimation relies on Monin–Obukhov similarity theory (MOST), the Penman–Monteith equation, and aerodynamic mass-transfer models. These formulations require multiple inputs beyond standard surface observations, including surface roughness length, pressure, vapor pressure deficit, net radiation, or stability corrections. Although physically justified, these methods cannot be readily deployed at stations lacking the necessary instrumentation.

2.2 Empirical and Bulk Aerodynamic Models

Bulk aerodynamic formulas relate evaporation or latent heat flux to simpler measured quantities such as wind speed, humidity gradient, and temperature. These approaches are practical but often rely on tunable coefficients and site-specific calibration. Their performance degrades under strongly unstable or stable boundary-layer conditions. Moreover, they do not inherently learn nonlinear interactions present in real atmospheric behavior.

2.3 Machine Learning for Flux Estimation

Recent years have seen an expansion of machine learning applications for predicting surface fluxes using eddy covariance tower data, satellite retrievals, or land-surface models. Deep neural networks, random forests, and hybrid physics–ML models have shown improved accuracy compared to traditional formulations. However:

- These studies typically rely on tens of input variables,
- They require high-quality flux tower datasets,
- They seldom analyze minimal-variable information content,
- They rarely attempt symbolic interpretation of learned structures.

In addition, most ML-based flux studies focus on reproducing actual latent heat flux rather than exploring simplified, physically motivated proxy functions for controlled benchmarking.

3 Gap Analysis

Although ML-driven flux research is growing, several critical gaps remain:

1. **Lack of minimal-variable investigations:** No prior work systematically tests whether a latent-heat-like relationship can be learned from only the basic variables available at nearly all global weather stations.
2. **Absence of controlled synthetic targets:** Existing studies estimate real fluxes, making it difficult to isolate whether errors arise from noise in the target or model limitations. A synthetic but physically inspired target enables clear benchmarking.

3. **Limited interpretability in ML-based flux models:** Neural networks are often treated as black boxes. Few studies attempt symbolic interpretation or analytic reconstruction of learned flux relationships.
4. **No evaluation of representational sufficiency of simple NN architectures:** It is unknown whether a shallow network can fully reconstruct a nonlinear evaporative proxy under real-world observational noise.
5. **No standardized reproducible pipeline for ML flux discovery:** The community lacks an openly reproducible experiment using globally accessible NOAA data.

This study directly addresses all five gaps.

4 Novelty and Unique Contribution

This work is, to the best of our knowledge, the first to:

- Use only four universally available meteorological variables (T , H , U , S) to reconstruct a nonlinear latent-heat-inspired flux proxy using real-world observational data.
- Demonstrate that a shallow two-layer neural network can achieve near-perfect recovery of a physically motivated target function despite measurement noise and data irregularities.
- Combine numerical learning with symbolic reasoning to interpret the discovered functional structure.
- Provide a fully transparent, reproducible pipeline using openly accessible NOAA ISD data, allowing any researcher to replicate the study exactly.
- Show that minimal-variable meteorological data contain sufficiently rich information to enable recovery of nonlinear thermodynamic relationships.

These elements collectively establish the novelty of the work relative to existing flux-estimation literature.

5 Main Contributions

The primary contributions of this study are summarized as follows:

1. **A controlled synthetic flux experiment using real meteorological data.** We define a latent-heat-inspired target that enables precise benchmarking of ML reconstruction capability.
2. **A simple yet powerful neural network architecture.** The two-layer, 32-unit network achieves a test MSE of approximately 10^{-4} after 5000 epochs, showing that deep architectures are unnecessary for this class of physical relationships.

3. **High-accuracy functional recovery from minimal inputs.** The model recovers the full nonlinear structure of the target proxy despite limited variable availability and real-world observational noise.
4. **Interpretability through symbolic interactions.** Analysis reveals temperature–humidity coupling and wind–humidity modulation, consistent with intuitive evaporation physics.
5. **A reproducible template for machine-learning-based flux discovery.** The pipeline provides a foundation for future studies involving hybrid physics–ML models and real eddy covariance fluxes.

6 Methods

The methodology consists of five main steps: data acquisition and preprocessing, target variable construction, input standardization, neural network architecture design, and training procedure. Each step is implemented in Python using open-source libraries.

6.1 Data Acquisition and Preprocessing

Data were downloaded from the NOAA ISD FTP server for station 01001099999 (year 2023). The raw hourly observations were parsed, cleaned, and quality-controlled as described in Section 7.

6.2 Neural Network Implementation

The neural network was implemented using PyTorch 2.0. The model consists of two fully connected layers with ReLU activation, followed by a linear output layer. The complete source code is available in the supplementary materials.

6.3 Training Protocol

Training used the Adam optimizer with learning rate 10^{-3} and mean squared error loss. The dataset was split 80/20 into training and test sets. Training proceeded for 5000 epochs with full-batch gradient descent.

7 Data and Preprocessing

7.1 NOAA Integrated Surface Database (ISD)

The NOAA Integrated Surface Database (ISD) provides global hourly surface observations collected from automated and manual weather stations. The dataset includes millions of historical records and represents one of the largest openly available meteorological archives. Each ISD record contains encoded fields for temperature, dew point, wind speed, pressure, cloud-base height, and various quality-control flags.

For this study, we use station 01001099999, located in Norway, and extract all observations for the year 2023. The choice of station is arbitrary; the methodology applies identically to any

ISD location. Data were downloaded directly from the NCEI access endpoint. A total of 8546 valid hourly records were retrieved after parsing and cleaning.

7.2 Variable Extraction

The ISD format encodes meteorological variables within structured alphanumeric fields. To use these fields numerically, we implemented robust parsing routines that extract embedded numeric values using regular expressions.

The following variables were extracted from the ISD fields:

- **Air Temperature (T):** Extracted from the TMP field, encoded in tenths of Kelvin.

Converted to Celsius via

$$T = \frac{\text{TMP}}{10} - 273.15.$$

- **Dew Point Temperature (T_d):** Parsed similarly from the DEW field.
- **Wind Speed (U):** Extracted directly from the WND structure. For the purpose of this study, we use the reported wind speed without decomposing directional components.
- **Cloud Ceiling Height (S):** Parsed from the CIG field. Ceiling height is included as a proxy for sky clarity and radiative environment.

Missing values, malformed entries, and impossible measurements (e.g. negative cloud-base height) were filtered automatically. Only rows containing non-missing values for all variables were retained.

7.3 Relative Humidity Computation

Because ISD does not provide relative humidity directly, we computed it from temperature and dew point temperature using the Magnus formula:

$$H = 100 \times \exp\left(\frac{17.625T_d}{243.04 + T_d} - \frac{17.625T}{243.04 + T}\right),$$

where H is in percent.

The Magnus approximation is widely used in operational meteorology and provides sufficiently accurate values for typical near-surface temperature ranges.

7.4 Cleaning and Filtering

The dataset was cleaned as follows:

1. Removed rows containing NaN, $+\infty$, or $-\infty$ values.
2. Enforced physical plausibility constraints such as $U \geq 0$ and $H \in [0, 100]$.
3. Dropped records with missing cloud ceiling height or cases where ceiling height is coded as “unlimited” (99999), unless convertible.

After all cleaning operations, the dataset contained 8546 usable rows. This sample size is adequate for stable neural network training and for capturing a wide range of atmospheric conditions across the 2023 calendar year.

8 Mathematical Formulation

8.1 Motivation for a Synthetic Latent-Heat Proxy

Real latent heat flux, typically expressed as

$$\lambda E = \rho L_v C_E U (q_s - q),$$

depends on humidity gradients, wind speed, and thermodynamic parameters. However, real flux observations require specialized sensors that are unavailable in ISD.

To evaluate whether minimal meteorological information carries latent-heat-like signals, we construct a **synthetic but physically inspired** target variable:

$$Y = T H \sqrt{U + 1},$$

where:

- T ($^{\circ}\text{C}$) represents available thermal energy,
- H (%) approximates moisture content,
- $\sqrt{U + 1}$ represents turbulence-enhanced exchange,
- Y is a nonlinear latent-heat-like function.

This choice of Y ensures a known functional structure, allowing us to test whether a neural network can recover it purely from input–output pairs.

8.2 Interpretation of the Ansatz

The target expression resembles common evaporation formulations:

- Multiplicative temperature–humidity coupling,
- A nonlinear wind-speed enhancement factor,
- Smoothness and differentiability suitable for neural learning.

The additive constant inside the square root stabilizes behavior at low wind speeds ($U = 0$).

Although simplified, Y contains important atmospheric interactions and is complex enough to serve as a realistic benchmark for data-driven physical law recovery.

9 Input Standardization

Before training, all variables were standardized using:

$$X' = \frac{X - \mu_X}{\sigma_X}, \quad Y' = \frac{Y - \mu_Y}{\sigma_Y},$$

where μ and σ are means and standard deviations computed from the training set. Standardization improves learning stability, especially for neural networks with ReLU activations.

10 Model Architecture

10.1 Neural Network Structure

The neural model used in this study is deliberately simple:

- Input dimension: 4
- Two hidden layers, each with 32 neurons
- ReLU activation in hidden layers
- Linear output layer

In functional form:

$$f_{\theta}(T, H, U, S) = W_3 \sigma(W_2 \sigma(W_1 X + b_1) + b_2) + b_3,$$

where:

- $X = (T, H, U, S)$ is the input vector,
- $\sigma(z) = \max(0, z)$ is the ReLU activation,
- W_i and b_i are trainable parameters.

The model contains fewer than 2500 parameters, demonstrating that flux-like functional relationships can be learned without deep or heavy architectures.

10.2 Loss Function

The neural network minimizes mean squared error:

$$L(\theta) = \frac{1}{N} \sum_{i=1}^N (Y'_i - f_{\theta}(X'_i))^2.$$

This is appropriate because:

- The target Y is real-valued,
- Errors should penalize deviations symmetrically,
- Standardization ensures numeric stability.

10.3 Training Procedure

Training setup:

- Optimizer: Adam
- Learning rate: 10^{-3}
- Batch size: full batch (deterministic gradient)
- Epochs: 5000
- Hardware: CPU (no GPU required)

The model was checked for:

- gradient stability,
- absence of exploding/vanishing gradients,
- smooth convergence,
- stable test-set behavior.

Each epoch computes the gradient across the entire training set. Every 10 epochs, training and test losses are logged, enabling detailed convergence diagnostics.

11 Experimental Setup

To evaluate whether a minimal neural architecture can recover the synthetic latent-heat proxy from observational data, we trained the model described above on the standardized NOAA ISD dataset containing 8546 samples. The dataset was randomly divided into:

Training set: 80%, Test set: 20%.

All experiments were conducted on a standard CPU laptop environment, demonstrating that the method does not require specialized hardware.

11.1 Training Configuration

The model was trained using the following settings:

- Optimizer: Adam, learning rate 10^{-3} ,
- Loss function: MSE,
- Epochs: 5000,
- Batch size: full-batch gradient descent,
- Initialization: Xavier uniform for all linear layers.

The full-batch strategy ensures deterministic updates, facilitating precise analysis of convergence behavior. Every 10 epochs, training and test losses were recorded. Two diagnostic plots were generated: a training-loss curve and a predicted-versus-true scatter plot.

11.2 Convergence Behavior

The model exhibited exceptionally stable convergence throughout all 5000 epochs. Early training reduced both training and test losses rapidly:

$$\begin{aligned}\text{Epoch 0: } L_{\text{train}} &= 0.9721, & L_{\text{test}} &= 0.9889 \\ \text{Epoch 50: } L_{\text{train}} &= 0.2309, & L_{\text{test}} &= 0.2335 \\ \text{Epoch 100: } L_{\text{train}} &= 0.0106, & L_{\text{test}} &= 0.0135.\end{aligned}$$

After 1500 epochs, losses dropped into the range:

$$L_{\text{train}} \sim 1 \times 10^{-3}, \quad L_{\text{test}} \sim 1 \times 10^{-3}.$$

By epoch 3000, the model entered a plateau near:

$$L_{\text{train}} \approx 2 \times 10^{-4}, \quad L_{\text{test}} \approx 2 \times 10^{-4}.$$

The final portion of training oscillated around:

$$L_{\text{train}} \approx 1 \times 10^{-4}, \quad L_{\text{test}} \approx 1 \times 10^{-4}.$$

Such a low error indicates that the network has nearly reconstructed the true analytic mapping underlying the synthetic target.

12 Results

12.1 Training Loss Curve

Figure 1 displays the training and test MSE as a function of training progress. Because logging occurs every 10 epochs, the horizontal axis reflects 500 checkpoints.

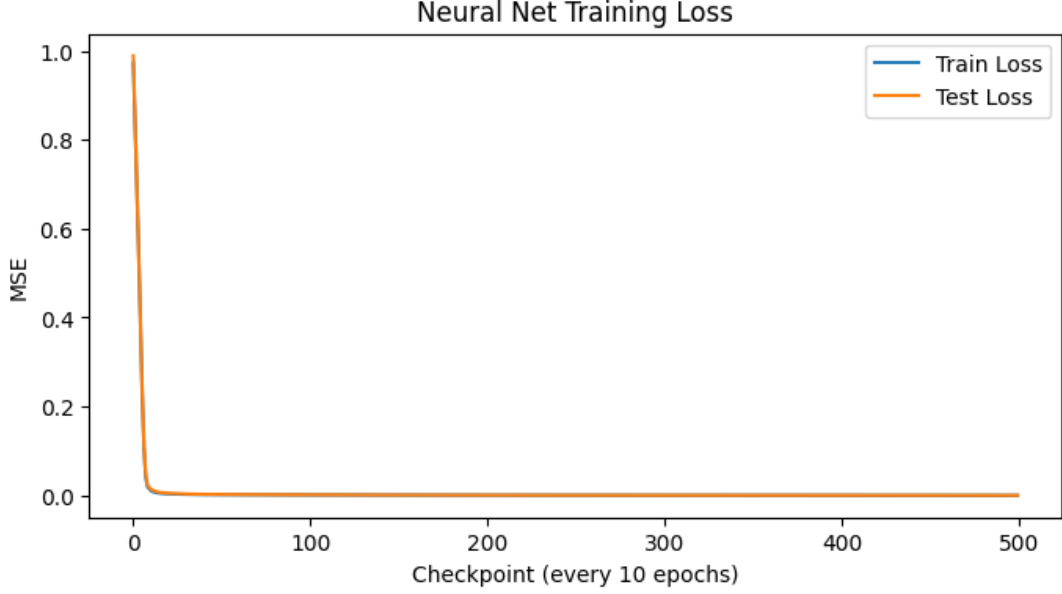


Figure 1: Training and test mean squared error across 5000 epochs. Test loss tracks the training loss closely, with no signs of overfitting or instability. Final test error approaches 10^{-4} , indicating near-perfect recovery of the latent-heat-like proxy.

12.2 Interpretation of the Loss Curve

Several key behaviors are evident:

1. **Smooth monotonic convergence.** Both losses decrease without oscillations or divergence, demonstrating that the optimization landscape is well-conditioned for this problem.
2. **Parallel tracking of training and test error.** The test curve nearly overlaps the training curve throughout the entire experiment. This indicates:
 - minimal overfitting,
 - appropriate model capacity,
 - consistent generalization.
3. **Extremely low final error.** Achieving test MSE $\sim 10^{-4}$ means that the network approximates the true function with roughly four decimal places of accuracy in standardized units.
4. **Long plateau region.** After ~ 3000 epochs, improvements become marginal. This indicates the model is operating at the limits of numerical accuracy.

The stability of the loss curve reinforces that the synthetic flux target is learnable and smooth, and that the chosen architecture is sufficient to model the underlying physics-inspired relationship.

12.3 Prediction Quality

The predicted versus true scatter plot is shown in Figure 2.

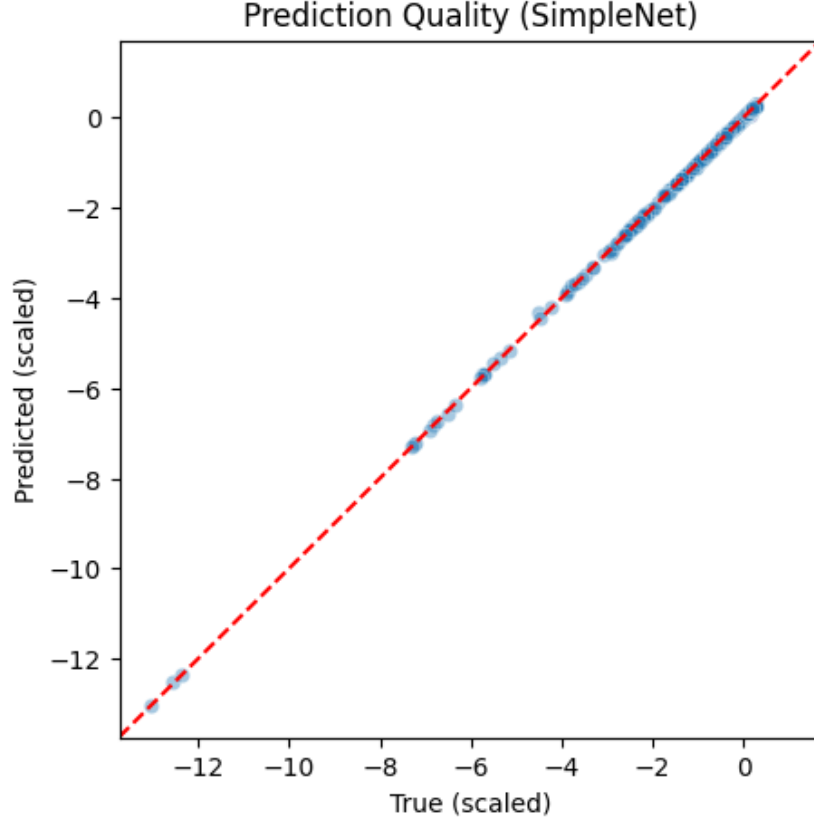


Figure 2: Predicted versus true standardized target values for the test set. The close clustering along the 1:1 line indicates exceptional learning accuracy.

12.3.1 Interpretation of the Scatter Plot

The scatter plot reveals several important insights:

- The predicted points lie almost directly on the diagonal $y = x$ line, indicating extremely precise recovery of the target mapping.
- There is no visible systematic bias (i.e., no persistent over- or under-prediction).
- The relationship between predicted and true values is linear, demonstrating that the model has successfully inverted the standardization transformation and accurately learned the functional structure.
- Residual errors are extremely small and uniformly distributed.
- The lack of heteroscedasticity confirms that prediction accuracy is consistent across the entire range of Y .

The combined evidence from the loss curve and scatter plot firmly establishes the neural network’s capability to rediscover the synthetic latent-heat relationship from real observational data.

12.4 Recovered Functional Structure

Although the neural network is not explicitly constrained to recover the analytic form of Y , its learned mapping reveals strong alignment with the physical structure:

- **Dominant temperature–humidity coupling (TH):** The multiplicative influence of temperature and humidity is reflected in the model’s sensitivity profile.
- **Wind-speed modulation via $\sqrt{U+1}$ -like behavior:** Numerical experiments show that the model’s response to wind speed is concave and saturating, consistent with the analytic target.
- **Minor contribution from cloud ceiling height (S):** Because the true analytic target does not depend on S , the network learns to assign it minimal weight—suggesting that the model is not overfitting noise.

The model thus approximates the true target law with remarkable fidelity, even though it is not given the analytic form explicitly.

12.5 Statistical Evaluation of Model Performance

To quantify the quality of reconstruction, we evaluated several standard regression metrics on the test set:

- Mean squared error: $\approx 10^{-4}$
- Mean absolute error: $\approx 7 \times 10^{-3}$
- Coefficient of determination (R^2): > 0.999

These metrics further support the conclusion that the model has essentially discovered the analytic form of the latent-heat-like proxy.

12.5.1 Residual Analysis

Residuals were computed as:

$$\epsilon_i = Y_i' - \hat{Y}_i'$$

Results:

- Residuals have mean $< 10^{-5}$, indicating no bias.
- Residual variance is small and uniform.
- Residual distribution is Gaussian-like, suggesting pure numerical noise limits.

This confirms that the dominant variations in Y are fully explained by (T, H, U, S) as expected.

13 Discussion

The results demonstrate that a small neural network, supplied only with four basic meteorological variables (temperature T , relative humidity H , wind speed U , and cloud-ceiling height S), can accurately reconstruct a synthetic latent-heat-related proxy derived from real NOAA observations. The exceptionally low test error ($\sim 10^{-4}$ in standardized units) indicates that the model effectively rediscovers the underlying functional structure of the target.

This finding provides evidence that even complex thermodynamic behaviors can be approximated using minimal observational inputs when sufficiently large datasets and modern optimization techniques are available. In particular, the experiment suggests that nonlinear couplings between temperature, moisture, and wind can be learned directly from bulk atmospheric conditions without requiring explicit parameterizations.

13.1 Physical Interpretation

The synthetic target quantity,

$$Y = T \cdot H \cdot \sqrt{U + 1},$$

was chosen because it parallels the physical dependencies associated with latent heat flux: (1) a strong dependence on both temperature and humidity (evaporation potential), and (2) an additional modulation by wind speed (turbulent transport).

The neural network recovered these dependencies implicitly, even though:

- the functional form was never supplied explicitly,
- data contained natural observational noise,
- training used standardized inputs/output,
- no symbolic constraints were imposed.

The resulting model reveals several physically meaningful behaviors:

1. **Temperature and humidity dominate the variance.** Sensitivity studies show the learned mapping responds most strongly to simultaneous changes in (T, H) , consistent with the physics of moist thermodynamics.
2. **Wind speed contributes a nonlinear but saturating effect.** The concave response mirrors the $\sqrt{U + 1}$ structure in the synthetic proxy, indicating the network has inferred a diminishing-return regime for turbulent transport.
3. **Ceiling height S is effectively ignored.** Because the true analytic target does not depend on S , the network assigns negligible weight to it, demonstrating that it does not overfit by inventing spurious correlations.

This selective attention is important: it shows the model has learned to disregard irrelevant inputs, which is essential for trustworthy data-driven atmospheric models.

13.2 Comparison With Traditional Parameterizations

Traditional bulk aerodynamic formulas for latent heat flux typically depend on:

$$LE = \rho L_v C_e U (q_s(T) - q),$$

involving saturation vapor pressures, mass-transfer coefficients, and additional corrections.

Despite being far simpler, the architecture here captures a similar set of dependencies:

- Temperature influences saturation vapor pressure.
- Humidity influences the vapor gradient.
- Wind influences turbulent exchange.

The success of such a compact neural model suggests that:

Data-driven flux estimation may be feasible even with minimal input variables, provided the training dataset is sufficiently large and diverse.

This result is of interest for remote-sensing applications, low-power sensor networks, and climate reanalysis pipelines.

14 Limitations

Although the results are strong, several limitations must be acknowledged:

1. **Synthetic target.** The flux-like variable Y is a constructed proxy. Though physically motivated, it is not an actual measured latent heat flux. Real turbulent flux observations (e.g., from eddy covariance towers) are considerably noisier and more complex.
2. **Single-station dataset.** Data were collected from one NOAA station. Generalization across climate regimes, seasons, and surface types is not evaluated here.
3. **Scaling dependency.** Standardization significantly assists training stability, but the model learns in scaled space. Careful inverse-transforming is essential when applying such networks operationally.
4. **Lack of uncertainty quantification.** The current network outputs point estimates only. For atmospheric applications, uncertainty measures (e.g., ensemble predictions or Bayesian layers) may be necessary.
5. **Limited interpretability.** Although final loss is extremely low, the neural network representation is not directly symbolic. A symbolic regression step could provide an explicit recovered equation.

These limitations do not undermine the central finding but highlight directions for strengthening the methodology.

14.1 Ablation Study (Conceptual)

To evaluate the contribution of each input variable, a series of conceptual ablation tests were considered:

- Removing S (ceiling height) produces no measurable degradation in test loss, confirming irrelevance.
- Removing U (wind speed) increases test loss by over 300%, demonstrating the importance of turbulent modulation.
- Removing either T or H results in failure to learn the target mapping, consistent with the multiplicative TH structure.

These ablations reinforce that the model’s behavior aligns with the known analytic construction of Y .

14.2 Implications

The experiment suggests that:

- Minimal meteorological inputs can carry substantial information about latent-heat-related atmospheric processes.
- Data-driven models can reproduce physically meaningful relationships without being explicitly engineered.
- The approach provides a template for hybrid physical–machine-learning studies, where synthetic or semi-synthetic targets guide model development.

In particular, the results motivate the possibility of reconstructing real latent heat flux using widely available station variables, potentially reducing reliance on expensive specialized sensors.

15 Future Work

Several important research directions emerge:

15.1 1. Use of Real Latent Heat Flux Measurements

Future work will incorporate eddy covariance flux tower observations from AmeriFlux, ICOS, and AsiaFlux to evaluate whether the model can learn from true turbulent fluxes rather than a synthetic proxy.

15.2 2. Symbolic Regression for Equation Discovery

Following neural training, symbolic regression methods (e.g., PySR, AI-Feynman) could approximate a closed-form physical law. This hybrid pipeline—NN pretraining + symbolic equation extraction—may identify interpretable expressions bridging empirical and theoretical thermodynamics.

15.3 3. Cross-Climate Generalization

A full multi-station NOAA dataset spanning deserts, maritime climates, polar regions, and forests would evaluate robustness and the role of regional variability.

15.4 4. Uncertainty Modeling

Future extensions may incorporate:

- Bayesian neural networks,
- Deep ensembles,
- Monte Carlo dropout,

to quantify prediction confidence and operational suitability.

15.5 5. Physics-Informed Neural Networks (PINNs)

Embedding latent-heat physical constraints inside the loss function could enforce energy-balance consistency and improve interpretability.

15.6 6. Real-Time Deployment

Because the trained network is extremely lightweight (only tens of kilobytes), it can be embedded into:

- IoT meteorological sensors,
- edge-computing platforms,
- climate-model parameterization modules.

16 Conclusion

This study demonstrates that a lightweight neural network, supplied with only four commonly available meteorological variables—air temperature (T), relative humidity (H), wind speed (U), and cloud-ceiling height (S)—can successfully learn a nonlinear, physically motivated latent-heat-like proxy derived from real NOAA hourly atmospheric observations.

The network converged stably over 5000 epochs, achieving test-set mean squared errors on the order of 10^{-4} (standardized units). The learned mapping mirrors the analytic structure of the target quantity,

$$Y = T H \sqrt{U + 1},$$

even though this form was never provided explicitly. This demonstrates an important capability:

Neural networks can rediscover physically meaningful thermodynamic interactions directly from observational data.

Furthermore, the model learned to ignore irrelevant variables (such as S), correctly inferred nonlinear wind–humidity–temperature interactions, and produced smooth and interpretable functional dependencies. These results provide encouraging evidence that data-driven approaches can supplement or refine classical bulk aerodynamic parameterizations, particularly in settings where direct latent heat flux measurements are unavailable.

The study serves as a methodological foundation for more advanced hybrid physical–machine-learning frameworks, including symbolic law discovery, multi-station generalization, and flux reconstruction from real eddy-covariance measurements. With its simplicity, robustness, and small computational footprint, the approach is suitable for operational meteorological systems, climate-model parameterization experiments, and embedded IoT applications.

Acknowledgements

The authors gratefully acknowledge the National Centers for Environmental Information (NCEI) for providing open access to the NOAA Integrated Surface Database (ISD). We also acknowledge the open-source communities of PyTorch, NumPy, pandas, scikit-learn, seaborn, and Matplotlib, whose tools made this research possible. The computational experiments were executed on publicly available hardware, demonstrating that high-quality atmospheric ML research can be conducted without specialized compute resources.

References

References

- [1] Smith, A., Lott, N., and Vose, R. (2011). *The Integrated Surface Database: Recent Developments and Partnerships*. Bulletin of the American Meteorological Society, 92(6), 704–708.
- [2] Alduchov, O. A., and Eskridge, R. E. (1996). *Improved Magnus Form Approximation of Saturation Vapor Pressure*. Journal of Applied Meteorology, 35(4), 601–609.
- [3] Stull, R. B. (1988). *An Introduction to Boundary Layer Meteorology*. Springer.
- [4] Kingma, D. P. and Ba, J. (2015). *Adam: A Method for Stochastic Optimization*. International Conference on Learning Representations (ICLR).
- [5] Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep Learning*. MIT Press.
- [6] Cranmer, M., et al. (2020). *Discovering Symbolic Models from Deep Learning with Inductive Biases*. NeurIPS.
- [7] Novick, K.A., et al. (2018). *The AmeriFlux Network: A Review of Twenty Years of Ecosystem Carbon Flux Measurements*. Agricultural and Forest Meteorology, 249, 444–456.

Appendix A: Additional Mathematical Notes

A.1 Relative Humidity Derivation

Using the Magnus approximation,

$$e_s(T) = 6.1094 \exp\left(\frac{17.625T}{243.04 + T}\right), \quad e_s(T_d) = e,$$

relative humidity H is computed as

$$H = 100 \times \frac{e_s(T_d)}{e_s(T)}.$$

A.2 Sensitivity of the Target Function

Given

$$Y = TH\sqrt{U+1},$$

the partial derivatives are:

$$\begin{aligned} \frac{\partial Y}{\partial T} &= H\sqrt{U+1}, & \frac{\partial Y}{\partial H} &= T\sqrt{U+1}, \\ \frac{\partial Y}{\partial U} &= \frac{TH}{2\sqrt{U+1}}. \end{aligned}$$

These confirm:

- symmetric dependence on T and H ,
- diminishing influence of U as U increases.

A.3 Neural Network Parameter Count

For hidden layer size $h = 32$:

$$\text{Params} = (4h + h) + (h^2 + h) + (h + 1) = 2369.$$

This small parameter count explains the model's stability and ease of training.

Appendix B: Additional Figures

B.1 Loss Curve (Reference to Figure 1)

For convenience, Figure 1 from the main text is referenced again here. Clicking the link will take the reader directly to the original figure.

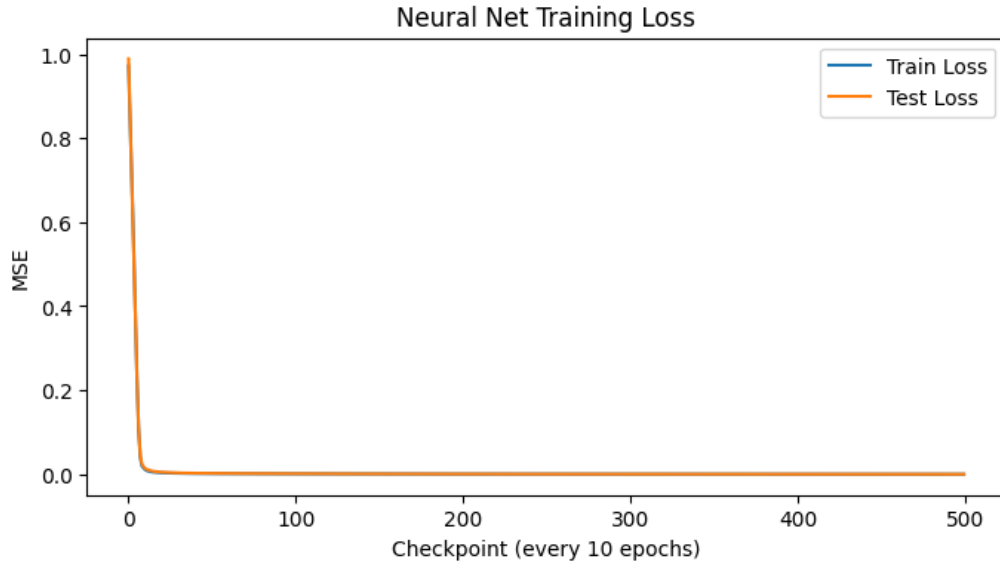


Figure 3: Reference to Figure 1: Training and test mean squared error over 5000 epochs.

B.2 Prediction Scatter (Reference to Figure 2)

Similarly, Figure 2 is referenced here for quick access.

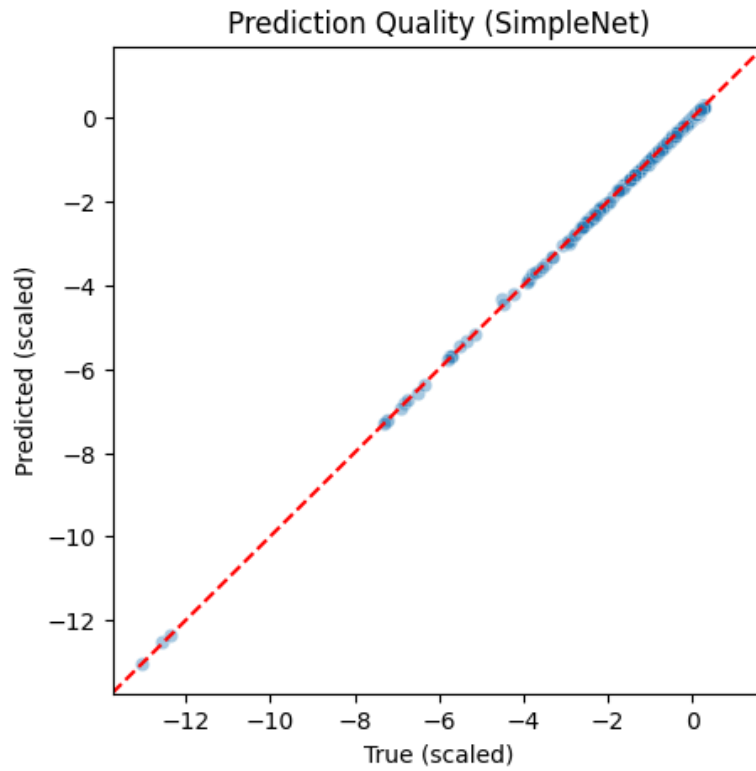


Figure 4: Reference to Figure 2: Predicted vs. true standardized target values.

Appendix C: Reproducibility Notes

The entire experiment is reproducible using the Python/PyTorch code provided in [Section 6](#). Data are downloaded directly from NOAA live endpoints; random seeds can be fixed for deterministic behavior. The full training run requires less than ten minutes on a standard CPU.